Academic subject: Computer Science			
Degree Class: L–35 – Scienze Matematiche		**Degree Course:** Mathematics	**Academic Year:** 2018/2019
		Kind of class: Mandatory	**Year:** 2 **Period:** 1
			ECTS: 6 divided into **ECTS lessons:** 4 **ECTS exe/lab/tutor:** 2

Time management, hours, in–class study hours, out–of–class study hours
lesson: 32   exe/lab/tutor: 16   in–class study: 48   out–of–class study: 102

Language: Italian	**Compulsory Attendance:** no		
Subject Teacher: Alessandro Bianchi	**Tel:** +390805442283 **e–mail:** alessandro.bianchi@uniba.it	**Office:** Department of Informatics Room 565, Floor V	**Office days and hours:** Wednesday 15.30 – 17.30 By appointment

Prerequisites:
Basic mathematics notions provided by Italian high schools

Educational objectives:
Learning basic concepts concerning programming methods and techniques; ability to apply them for solving problems by means of C language; basic notions and concepts about computability theory

Expected learning outcomes (according to Dublin Descriptors)	**Knowledge and understanding:** Learning fundamentals concepts for programming and computation **Applying knowledge and understanding:** Capability to apply learned concepts to simple algorithms programming and analysis **Making judgements:** Capability to judge the consistency of the logical structure used in building algorithms. Capability to identify the proper conceptual tools for solving programming problems. **Communication:** Learning the specific computer science language and methods for understanding textbooks, for explaining the learned knowledge, for describing, analyzing, and solving algorithmic problems. **Lifelong learning skills:** Acquiring suitable learning methods, supported by text consultation and by solving the exercises and questions periodically suggested during the course.

Course program
Introduction to the course: context, motivation and goal

Introduction to: algorithms; computers architecture; programming languages.

Introduction to programming:
    Decomposition techniques
    Representing algorithms and Structured programming
    Programming constructs
    Introduction to C

Introduction to computability
    Turing Machine
    Functions and computation
    Set enumeration and computation
    Recoursive functions

Church-Turing Thesis and Computability limitation
Introduction to computational complexity

Teaching methods:
Lectures and exercise sessions

Auxiliary teaching:
Didactic material available at url http://www.di.uniba.it/~bianchi/didattica/2017_18/inf_mat/index.htm

Assessment methods:
Exercises during the period, possibly exempting written exercises exam
Exam with (possible) written exercises and oral discussion

Bibliography:
M. Frixione, D. Palladino, Funzioni, Macchine, Algoritmi - Introduzione alla teoria della computabilitá, Carocci, 2004;
C. Toffalori, F. Corradini, S. Leonesi, S. Mancini, Teoria della computabilità e della complessità, McGraw-Hill, 2005
S. Ceri, D. Mandrioli, L. Sbattella, Informatica: Programmazione, McGraw-Hill, 2 Edizione, 2006;
H.M. Deitel, P.J. Deitel, C: Corso completo di programmazione, Apogeo, 2004;
L. Carlucci Aiello, F. Pirri, Strutture Logica Linguaggi, Pearson, 2005

Papers and slides cited / deployed during lectures.